

ERROR CORRECTION CODER

Publication number: JP61184930 (A)

Publication date: 1986-08-18

Inventor(s): NAKAMURA KATSUHIRO +

Applicant(s): NEC CORP +

Classification:

- international: H03M13/00; H03M13/00; (IPC1-7): H03M13/00

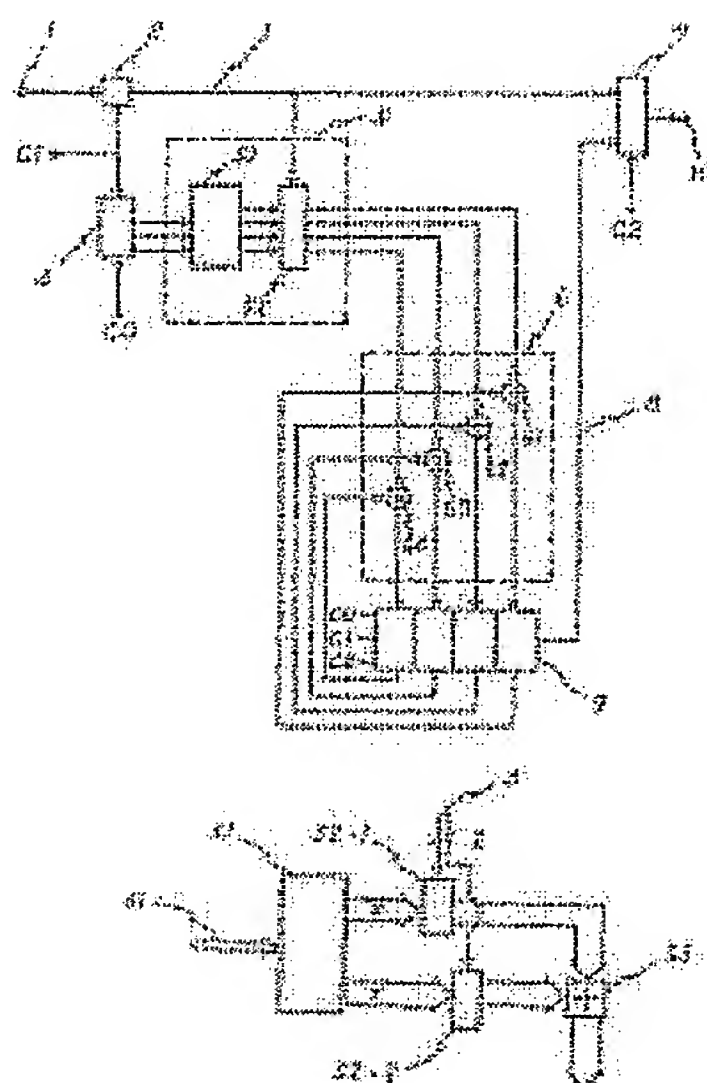
- European:

Application number: JP19850024401 19850213

Priority number(s): JP19850024401 19850213

Abstract of JP 61184930 (A)

PURPOSE:To realize easily a coder making a specific code word effective by providing a selecting circuit selecting and outputting an information digit string while it is being inputted and selecting and outputting contents of a register as a redundant digit string after the input of the information digit string is finished. CONSTITUTION:Both bit patterns X, Y are inhibited with information bit (0, 0), only a bit pattern X is inhibited with information bit (0, 1), only a bit pattern Y is inhibited with (1, 0) and a bit pattern from a read only memory 51 is given as it is in other cases. Each 4-bit bit pattern passing through gate circuits 52-1, 52-2 is subject to modulo 2 at each corresponding bit. The modulo 2 addition is subject to a modulo 2 addition combination circuit 53. Then a digit pattern converter 6 is constituted identically to a digit pattern converter in figure 2 and a data is fed from a register 7 to a redundant digit line in the unit of 2-bit. Through the constitution above, an error correction coder with code length of 12 and information bit of 8 can constitutes a coder with 2-bit parallel input/output.



Data supplied from the *espacenet* database — Worldwide

⑩ 日本国特許庁(JP)

⑪ 特許出願公開

⑫ 公開特許公報(A)

昭61-184930

⑬ Int. Cl.⁴
H 03 M 13/00

識別記号

庁内整理番号
6832-5J

⑭ 公開 昭和61年(1986)8月18日

審査請求 未請求 発明の数 1 (全7頁)

⑮ 発明の名称 誤り訂正符号器

⑯ 特 願 昭60-24401

⑰ 出 願 昭60(1985)2月13日

⑱ 発 明 者 中 村 勝 洋 東京都港区芝5丁目33番1号 日本電気株式会社内
⑲ 出 願 人 日本電気株式会社 東京都港区芝5丁目33番1号
⑳ 代 理 人 弁理士 岩佐 義幸

明細書

1. 発明の名称

誤り訂正符号器

2. 特許請求の範囲

(1) 入力される情報ディジット列に対し、この情報ディジット列に依存した冗長ディジット列を付加して出力する誤り訂正符号器において、

カウンタと、このカウンタのカウントに同期して順に入力される情報ディジットの列を、予め定めたルールで、前記情報ディジット列に依存して定めた第1のディジットパターンの列に変換するディジット列変換器と、このディジット列変換器から出力される前記第1のディジットパターンを、後記レジスタの内容に依存して、予め定めたルールで、第2のディジットパターンに変換するディジットパターン変換器と、前記第2のディジットパターンを格納する前記レジスタと、前記情報ディジット列が入力されてくる間は前記情報ディジット列を選択して出力し、前記情報ディジット列が入力され終わったあとは、前記レジスタの内容

を冗長ディジット列として選択し出力する選択回路とを具備することを特徴とする誤り訂正符号器。

3. 発明の詳細な説明

(産業上の利用分野)

本発明はディジタルデータの伝送あるいは蓄積などによって1ブロック内に生じた誤りを検出する、もしくは検出して訂正する誤り訂正符号の符号化装置に関する。

(従来技術)

説明の便宜上、データ伝送の場合に即して従来技術を説明する。

データ伝送における誤りは、伝送路上の雑音によるものが多いことが認められている。従来、そのような雑音の影響から逃れるために送信側では情報ビット列に冗長ビット列を付加して伝送路上に送り出し、受信側では、その冗長性をもとにこれを復号することによって誤りを検出し訂正するという方式を採用している。

この冗長ビット列を付加して誤り訂正符号を構成する方法として、従来一般によく知られ利用さ

れているものに、巡回符号を用いる方法がある。巡回符号についての詳細は、例えば、米国のマグローヒルブックカンパニー (McGRAW-HILL BOOK COMPANY) から1968年に発行された刊行物「アルジェブライックコーディングセオリー (Algebraic Coding Theory)」の8~20ページおよび119~144ページに詳しく述べられている。

この方法について簡潔に述べれば、情報ビット列 a_1, a_2, \dots, a_k に対応する冗長ビット列 $a_{k+1}, a_{k+2}, \dots, a_{k+m}$ は、次のようにして定められ、1ブロック内に生じた誤りを自動的に検出し訂正できるようになる。

まず、送信するビット列の長さ (符号長) を $N (=k+m)$ として、前記情報ビット列に対応する多項式 $I(x) = a_1 x^{N-1} + a_2 x^{N-2} + \dots + a_k x^{N-k}$ を予め定められた1又は0を係数とする m 次多項式 $g(x) = x^m + g_1 x^{m-1} + \dots + g_{m-1} x + 1$ で割り、剰余多項式 $R(x) = r_1 x^{m-1} + r_2 x^{m-2} + \dots + r_{m-1}$

$x + r_m$ を求める。この際の係数間の演算は2を法として行い、 $1+1=0+0=0$, $1+0=0+1=1$, $1 \cdot 1=1$, $0 \cdot 1=1 \cdot 0=0 \cdot 0=0$ とする。このとき、冗長ビット列 $a_{k+1}, a_{k+2}, \dots, a_{k+m}$ は、 $a_{k+1} = r_1$, $a_{k+2} = r_2, \dots, a_{k+m} = r_m$ として定められる。なお、多項式 $g(x)$ は、上記のようにして構成される誤り訂正符号の生成多項式と呼ばれる。

以上のような符号構成の仕方から、情報ビット列と、得られた冗長ビット列を係数ビット列とする符号多項式 $A(x)$ 、

$$A(x) = a_1 x^{N-1} + a_2 x^{N-2} + \dots + a_k x^{N-k} + a_{k+1} x^{N-k-1} + \dots + a_{k+m-1} x + a_{k+m}$$

は、生成多項式 $g(x)$ で割り切れるように構成されることになる。

一方、受信側では、このように構成された符号を受信しながら、この受信符号多項式を前記生成多項式で割り、その剰余多項式 $S(x)$ を求める。 $S(x)$ の係数パターンは、受信符号のシンδρο

ームと呼ばれる。ついで、 $S(x) = 0$ か否かを調べ、 $S(x) = 0$ ならば誤りなし、 $S(x) \neq 0$ ならば誤りありとする。通常、訂正すべき誤りビットパターンと剰余多項式 $S(x)$ とは1対1に対応するように生成多項式が選ばれているので、 $S(x)$ より誤りビットの位置が分り、この誤りビットが訂正される。もちろん、訂正なしで受信符号の誤りの検出だけにとどめる場合もある。

さて、上記のような誤り訂正符号を構成するための誤り訂正符号器としては、例えば前記刊行物の125ページあるいは128ページなどに具体的な回路のブロック図が記されてあるように、いわゆる符号多項式割算回路を利用するのが常であった。

(従来技術の問題点)

しかしながら、符号多項式割算回路を利用した符号器では、情報ビット列 a_1, a_2, \dots, a_k および冗長ビット列 $a_{k+1}, a_{k+2}, \dots, a_{k+m}$ の各ビットに対し、それぞれ単項式 $x^{N-1}, x^{N-2}, \dots, x, 1$ を対応づけてできる符号

多項式 $A(x)$ が生成多項式 $g(x)$ で割り切れるようにしか構成することができなかった。

またシステム構成の都合から、例えば j ビット短縮した符号 ($a_1, a_2, \dots, a_{k-j}, a_{k-j+1}, \dots, a_{k-j+m}$) を構成する場合にも、各ビットに対し、それぞれ単項式 $x^{N-j-1}, x^{N-j-2}, \dots, x, 1$ を対応づけてできる符号多項式が生成多項式 $g(x)$ で割り切れるように構成される。つまり先頭の $x^{N-1}, x^{N-2}, \dots, x^{N-j}$ が省略される形の符号多項式にしか、上記符号を対応づけられなかった。

ところが、応用によっては、上記各符号ビットと各単項式の対応づけを上記以外の形で、1対1に対応づけられるようにして符号化することが必要となる。例えば、変復調装置のキャリア位相の不確定性に対処するため、特定の符号語例えばすべて1の符号語 (1, 1, \dots , 1) が有効な符号語となるような短縮符号を構成したい場合や、あるいは、上記対応づけを、送受信者以外に対し秘密にすることによって、送信情報が第三者によ

って改ざんされたか否かの検証、つまりメッセージ認証をも、誤り訂正符号の冗長性に基づいて行おうとする場合である。

従来の誤り訂正符号化回路では、以上のような応用を実現し得ないという欠点があった。

〔発明の目的〕

本発明の目的は、従来の誤り訂正符号化回路の欠点を取り除き、新規な符号化法に基づく誤り訂正符号化回路を提供することにある。

〔発明の構成〕

本発明は、入力される情報ディジット列に対し、この情報ディジット列に依存した冗長ディジット列を付加して出力する誤り訂正符号器において、カウンタと、このカウンタのカウントに同期して順に入力される情報ディジットの列を、予め定めたルールで、前記情報ディジット列に依存して定めた第1のディジットパターンの列に変換するディジット列変換器と、このディジット列変換器から出力される前記第1のディジットパターンを、後記レジスタの内容に依存して、予め定めたル

らばすべて0ビットからなるmビットパターンを表す。又、 $a_1 B_1 + a_2 B_2$ をもって、 $a_1 B_1$ と $a_2 B_2$ との間に対応するディジット毎の演算+が施された結果得られるmディジットパターンを意味するものとする。例えば、 a_1 がバイナリーの場合は、 $a_1 B_1$ と $a_2 B_2$ との間に対応するビット毎の2を法とした加算を施して得られるmビットパターンを意味するものとする。このとき、mディジットの冗長ディジットパターン C_k は、

$$C_k = -(\dots (a_1 B_1 + a_2 B_2) + a_3 B_3 + \dots + a_k B_k)$$

で表される。ここで、mディジットパターン-Aは $A + (-A)$ がすべて0のmディジットパターンとなるようなディジットパターンである。

受信側では、受信符号語 ($a'_1, a'_2, \dots, a'_k, a'_{k+1}, a'_{k+2}, \dots, a'_{k+m}$) に対し、mディジットパターン

$$= (\dots ((a'_1 B_1 + a'_2 B_2) + a'_3 B_3 + \dots + a'_k B_k) + a'_{k+1} B_{k+1} + \dots + a'_{k+m} B_{k+m}) \dots (1)$$

ルで、第2のディジットパターンに変換するディジットパターン変換器と、前記第2のディジットパターンを格納する前記レジスタと、前記情報ディジット列が入力されてくる間は前記情報ディジット列を選択して出力し、前記情報ディジット列が入力され終わったあとは、前記レジスタの内容を冗長ディジット列として選択し出力する選択回路とを具備することを特徴としている。

〔発明の原理〕

本発明の原理は、次の通りである。情報ディジット列 a_1, a_2, \dots, a_k に対し、mビットの冗長ディジットパターン ($a_{k+1}, a_{k+2}, \dots, a_{k+m}$) を付加するものとする。情報ディジット列の各ディジットに対し、予め定められた各mディジットのパターン B_1, B_2, \dots, B_k が対応づけられているものとする。 $a_1 \cdot B_1$ をもって、 a_1, B_1 の間に予め定められた演算が施された結果得られるmディジットパターンを表す。例えば、 a_1 がバイナリーのビットの場合、 $a_1 = 1$ ならば B_1 そのものを、 $a_1 = 0$ なら

$$1 \cdot \dots + a'_{k+m} B_{k+m}) \dots (1)$$

を求める。但し、 B_{k+1} は、

$$(0, 0, \dots, 0, 1, 0, \dots, 0)$$

1

なるmディジットパターンである。

ついで、パターンDがすべて0のmディジットパターンとなるか否かを調べ、すべて0のmディジットパターンであれば誤りなし、すべて0のmディジットパターンでなければ、後に例で示すようにDをもとにして誤りディジットの位置と誤りの大きさを推定し訂正する。

本発明では、 $B_1, B_2, \dots, B_k, B_{k+1}, \dots, B_{k+m}$ が、任意のmディジットパターンとして前もって選べるように構成されている。

第1図は、本発明の基本構成図である。

図において、1は誤り訂正符号器の入力端子、2はレジスタ、3は情報ディジットライン、4はカウンタ、5はディジット列変換器、6はディジットパターン変換器、7はレジスタ、8は冗長ディジットライン、9はセレクト、10は誤り訂正

符号器の出力端子である。

入力端子1から入力された情報ディジット a_1, a_2, \dots, a_k の列は、一旦レジスタ2にストアされたのち、情報ディジットライン3を介してディジット列変換器5に送られ、第1のディジットパターンの列 B'_1, B'_2, \dots, B'_k に変換される。その際、入力である情報ディジット列も、出力であるディジットパターンの列も、カウンタ4のカウンタ(1, 2, \dots , k)に同期して入出力される。カウンタのカウンタ i におけるディジットパターン B'_i は、カウンタ i におけるレジスタ7の内容に依存して、ディジットパターン変換器6によって第2のディジットパターン C_i に変換されて、レジスタ7に格納される。最後の情報ディジット a_k が入力され終わったあと、第2のディジットパターン C_k がレジスタ7に格納される。この C_k が冗長データパターン($a_{k+1}, a_{k+2}, \dots, a_{k+m}$)であり、冗長ディジットライン8を介して、セクタ9に供給される。セクタ9では、情報ディジット列 a_1, a_2, \dots, a_k が情報ディジットライン3を介してセクタに供給される間は、これら情報ディジット列 a_1, a_2, \dots, a_k をセレクトし、そのあとは、冗長ディジットライン8を介して供給される冗長ディジット列 $a_{k+1}, a_{k+2}, \dots, a_{k+m}$ をセレクトして出力端子10へセレクトしたディジット列を出力する。

〔実施例〕

一実施例として、本発明に従って構成した符号長12、情報ビット数8の誤り訂正符号器を第2図に示す。第1図と同一の機能を有するブロックないしラインには、同一の番号を付して示している。

この実施例においては、ディジット列変換器5は、例えば第1表に示すビットパターンが格納されているリードオンリメモリ(またはランダムアクセスメモリ)51と、対応する情報ビットが0のときは、リードオンリメモリ51の出力ビットパターンのビットパターン変換器6への供給をインヒビットし、対応する情報ビットが1のときは、

リードオンリメモリ51の出力ビットパターンをそのままの形でビットパターン変換器6への供給を許可するゲート回路52とから構成されている。また、ビットパターン変換器6は、4個のモジュロ2加算器61~64で構成されている。これら加算器の一方の入力端子はゲート回路52の並列出力端子にそれぞれ接続され、出力端子はレジスタ7の並列入力端子にそれぞれ接続されている。レジスタ7の並列出力端子はモジュロ2加算器の他方の入力端子にそれぞれ接続されている。

第1表

アドレス	ビットパターン	
0	1 0 1 1	B_0
1	1 1 1 1	B_1
2	0 1 0 1	B_2
3	1 0 1 0	B_3
4	1 1 0 1	B_4
5	0 0 1 1	B_5
6	0 1 1 0	B_6
7	1 1 0 0	B_7
	0 0 0 1	B_8
	0 0 1 0	B_9
	0 1 0 0	B_{10}
	1 0 0 0	B_{11}

第3図は、第2図に示す誤り訂正符号器の動作に用いられるリセットパルス C_0 、クロックパルス C_1 および制御パルス C_2 の波形を示す。リセットパルス C_0 はカウンタ4およびレジスタ7に供給され、クロックパルス C_1 はレジスタ2, 7

およびカウンタ4に供給され、制御パルスC2はレジスタ7およびセクタ9に供給される。

第2図に示す誤り訂正符号器の入力端子1に情報ビット列 a_1, a_2, \dots, a_7 が入力されると、先頭の情報ビット a_1 がレジスタ2にストアされると同時に、リセットパルスC0にてカウンタ4およびレジスタ7はリセットされる。ディジット列変換器5のリードオンリメモリ51内には、第1表に示したビットパターンが格納されており、カウント数をリードオンリメモリ51のアドレスとして与える。カウント i のときリードオンリメモリ51の出力ビットパターン B_i は、対応する情報ビット a_i が0のときはゲート回路52でインヒビットされ1のときはそのままの形でビットパターン変換器6へ供給される。

ビットパターン変換器6は、前述したようにモジュロ2加算器61~64で構成されており、従って例えば $a_0 = a_1 = 1, a_2 = a_3 = a_4 = a_5 = a_6 = a_7 = 0$ のときは、レジスタ7に最終的に $(1011) \oplus (1111) = (0100$

ン B_5 つまり (0011) となる。一般にビット a_i が間違っていると前記(1)式の結果は、第1表に示したビットパターン B_i となる。 B_0 から B_{11} までのビットパターンはすべて相異なるので、得られた演算結果をみて何番目のビットにエラーを生じたかを推定し訂正することができる。

一方、 $a_0 = a_1 = a_2 = a_3 = a_4 = a_5 = a_6 = a_7 = 1$ のとき第1表より、冗長ビット列は1, 1, 1, 1となる。つまりすべて1のワード $(1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1)$ も有効な符号語となっている。このことは前述したように、本発明に従えば、変復調装置のキャリア位相の不確定性に対処できる誤り訂正符号が構成できることを意味している。これに対し、符号長12の誤り訂正符号で、従来のように符号多項式割算回路を用いる方法では、すべて1のワードを有効な符号語とすることはできないので、上記キャリア位相の不確定性に対処できる誤り訂正符号は形成できない。

なお、リードオンリメモリ51に第1表で示し

が格納され、冗長ビット列0, 0, 1, 0が直列出力端子から情報ディジットライン8を介してセクタ9に供給される。レジスタ7は、第3図に示した制御パルスC2によって、レジスタの内容をパラレルにビットパターン変換器6へ供給するかあるいはビットパターン変換器6からレジスタ7へのビットパターンのセットをインヒビットした状態で、レジスタ7の内容をセクタ9へ供給するかを制御される。また、セクタ9が情報ビット列を選択するか冗長ビット列を選択するかも制御パルスC2によって制御される。

以上のように例えば $a_0 = a_1 = 1, a_2 = a_3 = a_4 = a_5 = a_6 = a_7 = 0$ のときの冗長ビットは、 $a_8 = 0, a_9 = 0, a_{10} = 1, a_{11} = 0$ となる。受信側で、これら情報ビット列および冗長ビット列よりなる誤り訂正符号に対し、前記(1)式を求めると確かにビットパターン $(0, 0, 0, 0)$ が得られるが、例えばビット a_5 が伝送路上でエラーを起こし、0から1になったとすれば、前記(1)式の演算結果はビットパター

たようなビットパターンの表をいくつか用意し、どの表を選ぶかを送受信者間の秘密にしておけば、伝送路上で第三者が故意に伝送内容を改ざんしたとしても、符号自体の誤り検出能力でもって受信者には容易に検知されるし、しかも、逆に言って、第三者は容易には、伝送路上の情報を改ざんできない。なぜなら、符号化ルールが分からないからである。

次に、前記符号長12、情報ビット数8の誤り訂正符号を、例えば2ビット並列処理する場合を考えてみる。

第1図において、情報ディジットは、 $A_0 = (a_0, a_1), A_1 = (a_2, a_3), A_2 = (a_4, a_5), A_3 = (a_6, a_7)$ の2ビット単位で入力端子1に与えられるものとする。レジスタ2も2ビット分のレジスタとする。カウンタ4は、0, 1, 2, 3とカウントする。ディジット列変換器5は、例えば第4図の如く構成される。第4図において、アドレス入力ライン41を介して入力されたアドレスに格納してあるビットパタ

ーンを、リードオンリメモリ51は出力する。リードオンリメモリ51の内容は、第2表に示す通りである。

第2表

アドレス	ビットパターン		
	X	Y	B _i
0	1 0 1 1	1 1 1 1	B ₀
1	0 1 0 1	1 0 1 0	B ₁
2	1 1 0 1	0 0 1 1	B ₂
3	0 1 1 0	1 1 0 0	B ₃

第2表において、アドレス*i*の内容(X, Y)は、第1表におけるアドレス2*i*の内容(=X)とアドレス(2*i*+1)の内容(=Y)とから成る。そこで、第2図の誤り訂正符号器と同一の機能を持たせるために、第3図のディジット列変換器では、ゲート回路52-1と52-2を備えて、情報ディジットライン3から入力されてくる2ビット分の情報ビット(x, y)によって上記ビットパターン(X, Y)をゲートする。つまり情

報ビット(0, 0)のときはビットパターンX, Y両方とも、また、情報ビット(0, 1)のときはビットパターンXのみを、(1, 0)のときはビットパターンYのみをインヒビットし、その他の場合は、リードオンリメモリ51からのビットパターンをそのまま通す。ゲート回路52-1, 52-2を通り抜けた各々4ビットのビットパターンを対応するビット毎にモジュロ2加算する。このモジュロ2加算は、モジュロ2加算組合せ回路53にて行われる。ついで、ディジットパターン変換器6は、第2図のディジットパターン変換器と同じにし、冗長ディジットラインへは、レジスタ7から2ビット単位で供給するように構成する。このような構成に従えば、符号長12, 情報ビット数8の誤り訂正符号器で、2ビット並列で入出力される符号器も構成できることがわかる。

以上、本発明の実施例を説明したが本発明はこれら実施例に限定されるものではなく本発明の範囲内で種々の変形、変更が可能なことは勿論である。

〔発明の効果〕

以上述べてきたように、本発明に従えば、従来の符号多項式割算回路を用いる場合と異なり、特定の符号語(例えば、すべて1の符号語(1, 1, ..., 1))を有効な符号語として含むような誤り訂正符号の符号器を、符号長の如何にかかわらず、容易に構成できる。

更には、メッセージ認証の機能を兼ねた符号器も容易に構成できる。

これらは、今後のディジタル通信回路網における符号構成上、多大の効果を発揮できることは明らかである。

4. 図面の簡単な説明

第1図は本発明の基本構成図、

第2図は本発明の一実施例を示す図、

第3図は第2図の実施例の動作に用いられるパルスの波形を示す図、

第4図は第1図におけるディジット列変換器の一構成例を示す図である。

1 入力端子

2 レジスタ

3 情報ディジットライン

4 カウンタ

5 ディジット列変換器

6 ディジットパターン変換器

8 冗長ディジットライン

9 セレクタ

10 出力端子

51 リードオンリメモリ

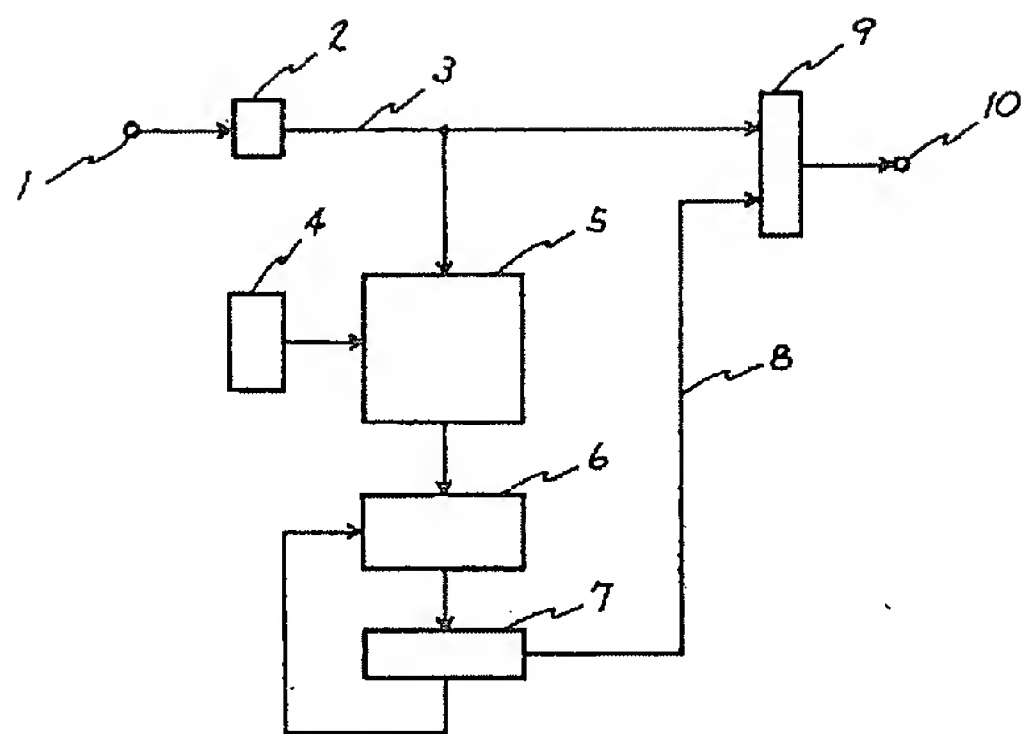
52-1, 52-2 ゲート回路

61, 62, 63, 64

. モジュロ2加算器

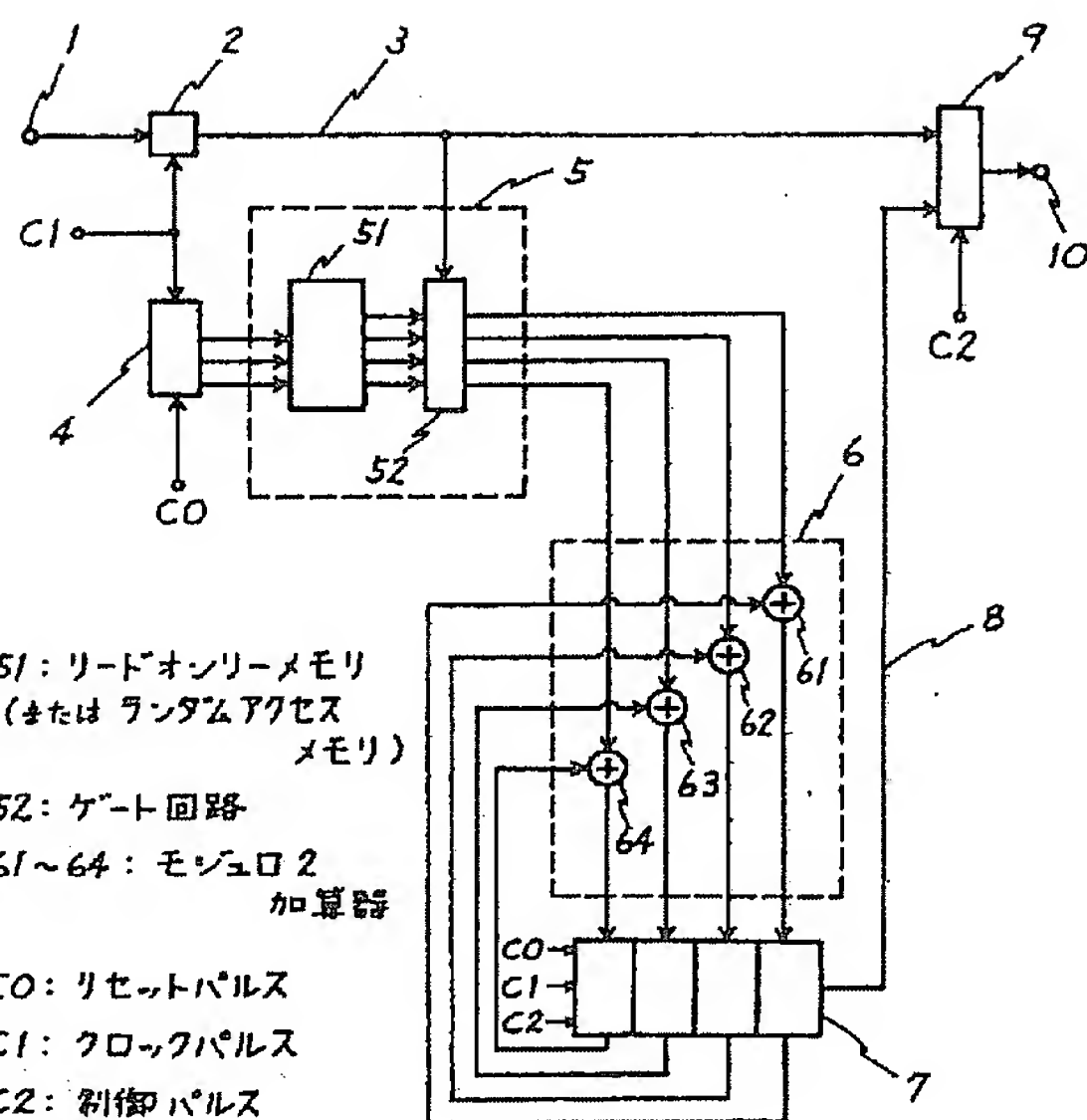
53 モジュロ2加算組合せ回路

代理人弁理士 岩佐 義幸



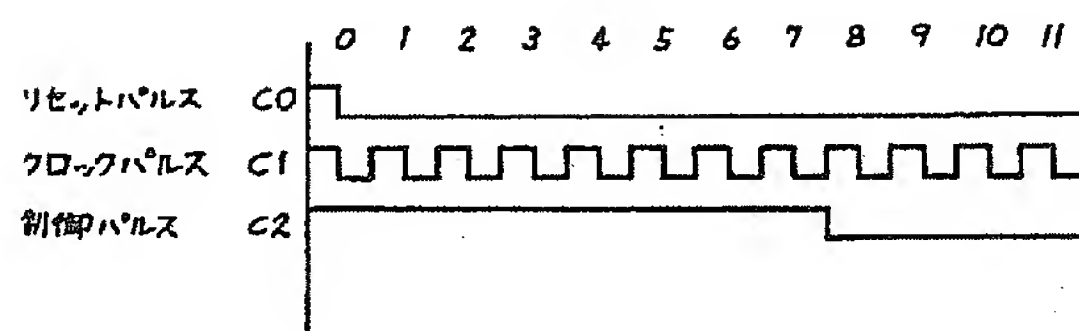
- 1: 入力端子
2: レジスタ
3: 情報ディジットライン
4: カウンタ
5: デジタリ列変換器
6: デジタリパターン変換器
7: レジスタ
8: 冗長ディジットライン
9: セレクタ
10: 出力端子

第 1 図

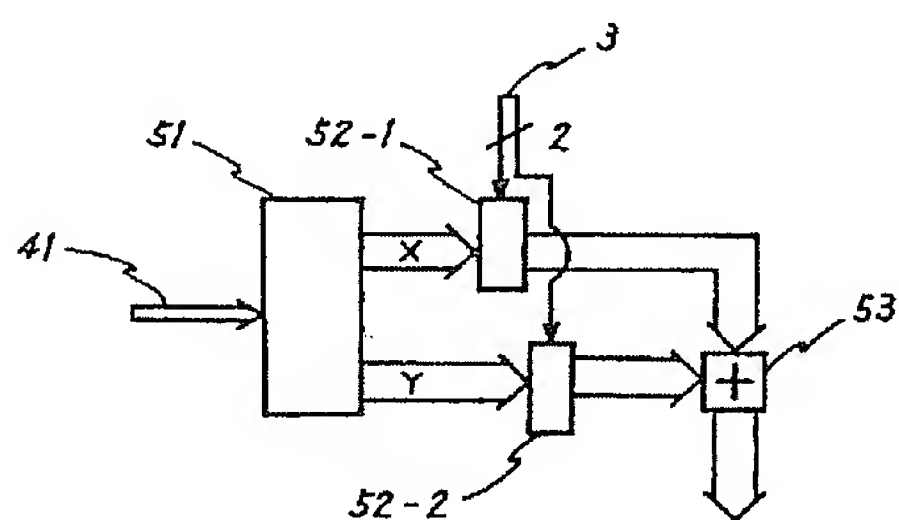


- 51: リードオンリーメモリ
(または ランダムアクセス
メモリ)
52: ゲート回路
61~64: モジュロ2
加算器
C0: リセットパルス
C1: クロックパルス
C2: 制御パルス

第 2 図



第 3 図



- 41: アドレス入力ライン
52-1, 52-2: ゲート回路
53: モジュロ2 加算組合せ回路

第 4 図